On Calibrating Diffusion Probabilistic Models

Zhijie Deng

zhijied@sjtu.edu.cn

Shanghai Jiao Tong University

Joint work with Tianyu Pang, Cheng Lu, Chao Du, Min Lin, and Shuicheng Yan

Diffusion Models in 2020 (Nonequilibrium Thermodynamics)

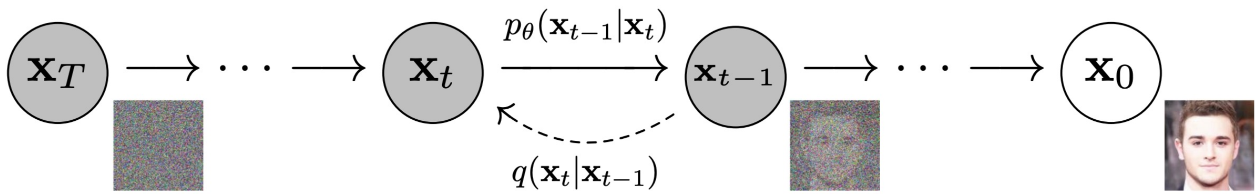

Figure 2: The directed graphical model considered in this work.

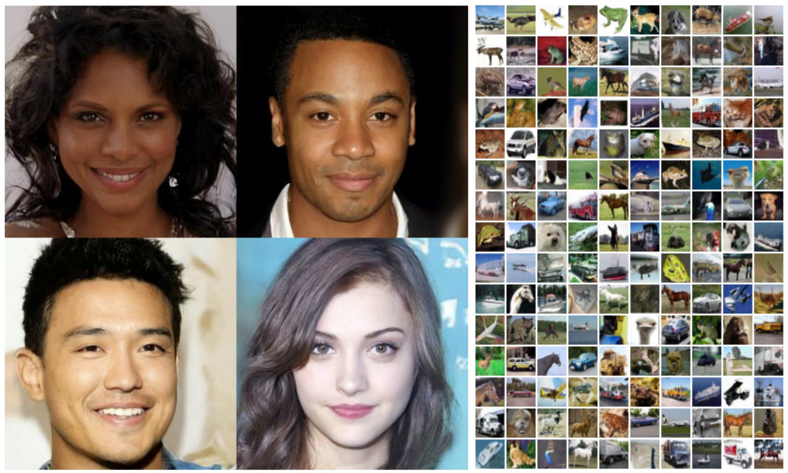

Figure 1: Generated samples on CelebA-HQ 256 × 256 (left) and unconditional CIFAR10 (right)

Figure 3: LSUN Church samples. FID=7.89

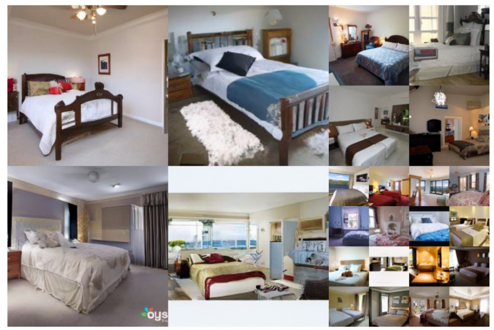

Figure 4: LSUN Bedroom samples. FID=4.90

[1] Sohl-Dickstein et al. Deep Unsupervised Learning using Nonequilibrium Thermodynamics. ICML 2015
[2] Ho et al. Denoising Diffusion Probabilistic Models. NeurIPS 2020

Diffusion Models in 2020 (Annealed Langevin Dynamics)

Algorithm 1 Annealed Langevin dynamics.

Require: $\{\sigma_i\}_{i=1}^{L}, \epsilon, T.$

Initialize $\tilde{\mathbf{x}}_0$

for $i \leftarrow 1$ to L **do**

    $\alpha_i \leftarrow \epsilon \cdot \sigma_i^2 / \sigma_L^2$ $\quad \triangleright$ α_i is the step size.

    for $t \leftarrow 1$ to T **do**

        Draw $\mathbf{z}_t \sim \mathcal{N}(0, I)$

        $\tilde{\mathbf{x}}_t \leftarrow \tilde{\mathbf{x}}_{t-1} + \frac{\alpha_i}{2}\mathbf{s}_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}_{t-1}, \sigma_i) + \sqrt{\alpha_i}\, \mathbf{z}_t$

    end for

    $\tilde{\mathbf{x}}_0 \leftarrow \tilde{\mathbf{x}}_T$

end for

return $\tilde{\mathbf{x}}_T$

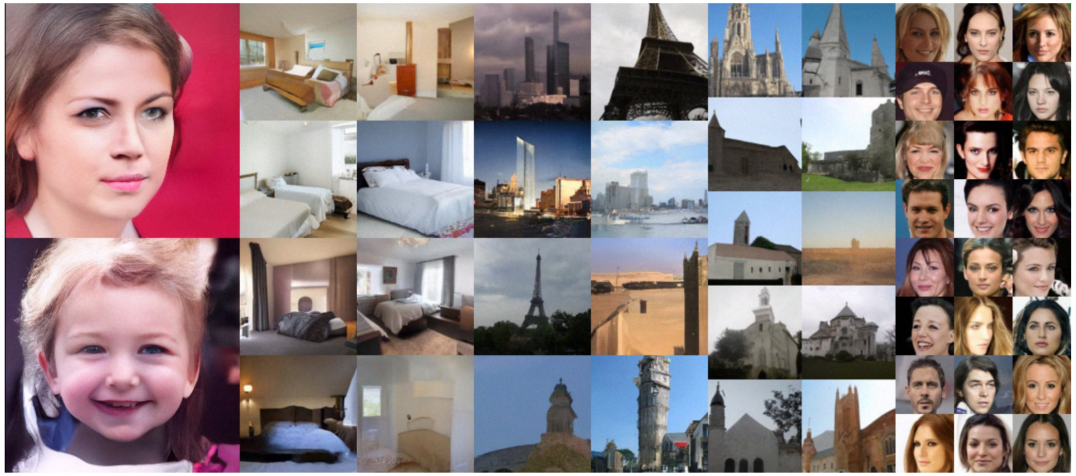

Figure 1: Generated samples on datasets of decreasing resolutions. From left to right: FFHQ 256×256, LSUN bedroom 128×128, LSUN tower 128×128, LSUN church_outdoor 96×96, and CelebA 64×64.

EBMs (BP through CNNs) → Score-based models (U-Nets)

[3] Song & Ermon. Generative Modeling by Estimating Gradients of the Data Distribution. NeurIPS 2019
[4] Song & Ermon. Improved Techniques for Training Score-Based Generative Models. NeurIPS 2020

Diffusion Models in 2021 (Stochastic Differential Equations)

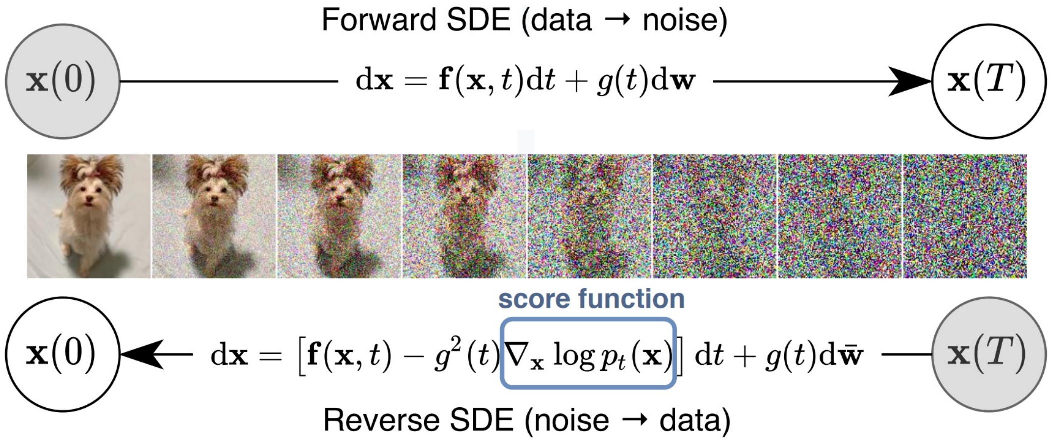

Figure 1: **Solving a reverse-time SDE yields a score-based generative model.** Transforming data to a simple noise distribution can be accomplished with a continuous-time SDE. This SDE can be reversed if we know the score of the distribution at each intermediate time step, $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$.

- Drift coefficient f
- Diffusion coefficient g

[5] Song et al. Score-Based Generative Modeling through Stochastic Differential Equations. ICLR 2021

Diffusion Processes

Forward process (transition distribution):

$$x_0 \sim q_0(x_0), \qquad q_{0t}(x_t|x_0) = \mathcal{N}(x_t|\alpha_t x_0, \sigma_t^2 \mathbf{I})$$

Forward process (SDE):

$$dx_t = f(t)x_t dt + g(t)d\omega_t$$

where $f(t) = \dfrac{d\log\alpha_t}{dt}$ and $g(t)^2 = \dfrac{d\sigma_t^2}{dt} - 2\dfrac{d\log\alpha_t}{dt}\sigma_t^2$

Diffusion Processes

Reverse process (SDE):

$$dx_t = \left[f(t)x_t - g(t)^2 \nabla_{x_t} \log q_t(x_t)\right] dt + g(t) d\overline{\omega}_t$$

Reverse process (ODE):

$$\frac{dx_t}{dt} = f(t)x_t - \frac{1}{2} g(t)^2 \nabla_{x_t} \log q_t(x_t)$$

Training DPMs by Score Matching

$$\mathcal{J}_{\mathrm{SM}}^{t}(\theta) \triangleq \frac{1}{2}\mathbb{E}_{q_t(x_t)}\left[\|\boldsymbol{s}_\theta^t(x_t) - \underbrace{\nabla_{x_t}\log q_t(x_t)}_{\text{Unknown}}\|_2^2\right]$$

$$\mathcal{J}_{\mathrm{SM}}(\theta;\lambda(t)) \triangleq \int_0^T \lambda(t)\mathcal{J}_{\mathrm{SM}}^{t}(\theta)dt$$

Training DPMs by Denoising Score Matching

$$\mathcal{J}_{\mathrm{DSM}}^{t}(\theta) \triangleq \frac{1}{2}\mathbb{E}_{q_0(x_0),q(\epsilon)}\left[\left\|\boldsymbol{s}_\theta^t(x_t) + \frac{\epsilon}{\sigma_t}\right\|_2^2\right]$$

where $x_t = \alpha_t x_0 + \sigma_t \epsilon$ and $q(\epsilon) = \mathcal{N}(\epsilon|\mathbf{0}, \mathbf{I})$

The Stochastic Process of Data Score is a Martingale

Theorem 1. *(Proof in Appendix A.1) Let $q_t(x_t)$ be constructed from the forward process in Eq. (2). Then under some regularity conditions, we have $\forall 0 \leq s < t \leq T$,*

$$\alpha_t \nabla_{x_t} \log q_t(x_t) = \mathbb{E}_{q_{st}(x_s|x_t)} \left[\alpha_s \nabla_{x_s} \log q_s(x_s) \right], \tag{6}$$

where $q_{st}(x_s|x_t) = \frac{q_{st}(x_t|x_s)q_s(x_s)}{q_t(x_t)}$ is the transition probability from x_t to x_s.

Leads to concentration bounds and naturally $\mathbb{E}_{q_t(x_t)} \left[\nabla_{x_t} \log q_t(x_t) \right] = 0$

Calibrating DPMs

Although $\mathbb{E}_{q_t(x_t)}\left[\nabla_{x_t} \log q_t(x_t)\right] = 0$

Typically there is $\mathbb{E}_{q_t(x_t)}\left[\boldsymbol{s}_\theta^t(x_t)\right] \neq 0$

So we calibrate DPMs into $\boldsymbol{s}_\theta^t(x_t) - \eta_t$

Calibrating DPMs

Given any pretrained DPM, we can calibrate it as:

$$\boldsymbol{s}_\theta^t(x_t) - \mathbb{E}_{q_t(x_t)}\left[\boldsymbol{s}_\theta^t(x_t)\right]$$

$$\mathcal{J}_{\mathrm{SM}}^t(\theta, \eta_t^*) = \mathcal{J}_{\mathrm{SM}}^t(\theta) - \frac{1}{2}\left\|\mathbb{E}_{q_t(x_t)}\left[\boldsymbol{s}_\theta^t(x_t)\right]\right\|_2^2$$

Likelihood of Calibrating DPMs: *SDE Solver*

$$dx_t = \left[f(t)x_t - g(t)^2 \boldsymbol{s}_\theta^t(x_t)\right] dt + g(t)d\overline{\omega}_t$$

Marginal distribution

$$p_t^{\mathrm{SDE}}(x_t;\theta)$$

$$\mathcal{D}_{\mathrm{KL}}\left(q_0 \| p_0^{\mathrm{SDE}}(\theta)\right) \leq \mathcal{J}_{\mathrm{SM}}(\theta; g(t)^2) + \mathcal{D}_{\mathrm{KL}}\left(q_T \| p_T\right)$$

Likelihood of Calibrating DPMs: *SDE Solver*

$$dx_t = \left[f(t)x_t - g(t)^2(\boldsymbol{s}_\theta^t(x_t) - \underline{\eta_t})\right]dt + g(t)d\overline{\omega}_t$$

$$\Downarrow \text{ Marginal distribution}$$

$$p_0^{\mathrm{SDE}}(x_t; \theta, \underline{\eta_t})$$

$$\mathcal{D}_{\mathrm{KL}}\left(q_0 \| p_0^{\mathrm{SDE}}(\theta, \underline{\eta_t})\right) \leq \mathcal{J}_{\mathrm{SM}}(\theta, \underline{\eta_t}; g(t)^2) + \mathcal{D}_{\mathrm{KL}}\left(q_T \| p_T\right)$$

Likelihood of Calibrating DPMs: *SDE Solver*

$$\mathcal{J}_{\mathrm{SM}}(\theta, \eta_t^*; g(t)^2) = \mathcal{J}_{\mathrm{SM}}(\theta; g(t)^2) - \underline{\frac{1}{2}\int_0^T g(t)^2 \left\| \mathbb{E}_{q_t(x_t)} \left[\boldsymbol{s}_\theta^t(x_t) \right] \right\|_2^2 dt}$$

Upper bound reduced by calibration

Likelihood of Calibrating DPMs: *ODE Solver*

$$\frac{dx_t}{dt} = f(t)x_t - \frac{1}{2}g(t)^2 \boldsymbol{s}_\theta^t(x_t)$$

$\Downarrow$ Marginal distribution

$$p_t^{\mathrm{ODE}}(x_t;\theta)$$

$$\mathcal{D}_{\mathrm{KL}}\left(q_0 \| p_0^{\mathrm{ODE}}(\theta)\right) \approx \mathcal{J}_{\mathrm{SM}}(\theta; g(t)^2) + \mathcal{D}_{\mathrm{KL}}\left(q_T \| p_T\right)$$

Likelihood of Calibrating DPMs: *ODE Solver*

$$\frac{dx_t}{dt} = f(t)x_t - \frac{1}{2}g(t)^2(\boldsymbol{s}_\theta^t(x_t) - \underline{\eta_t})$$

$\Downarrow$ Marginal distribution

$$p_t^{\mathrm{ODE}}(x_t; \theta, \underline{\eta_t})$$

$$\mathcal{D}_{\mathrm{KL}}\left(q_0 \| p_0^{\mathrm{ODE}}(\theta, \underline{\eta_t})\right) \approx \mathcal{J}_{\mathrm{SM}}(\theta, \underline{\eta_t}; g(t)^2) + \mathcal{D}_{\mathrm{KL}}\left(q_T \| p_T\right)$$

Empirical Results

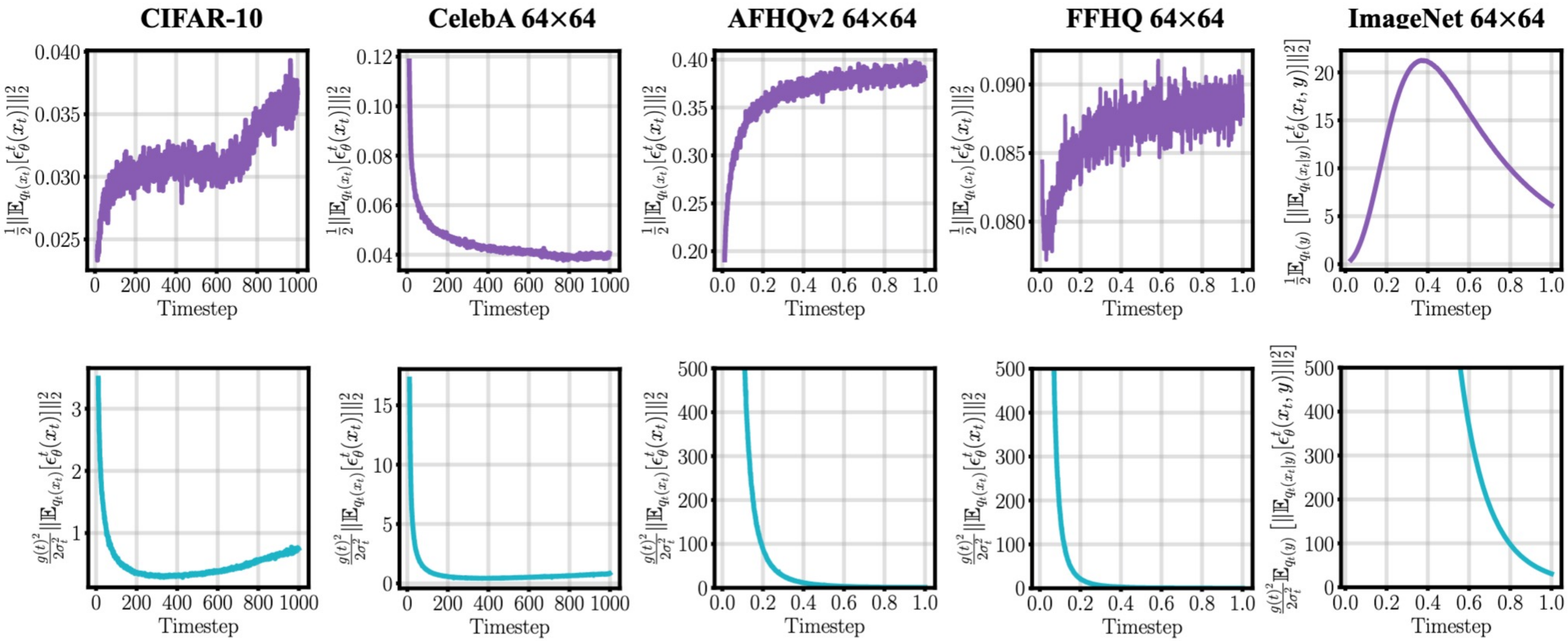

Figure 1: Time-dependent values of $\frac{1}{2}\|\mathbb{E}_{q_t(x_t)}\left[\boldsymbol{\epsilon}_\theta^t(x_t)\right]\|_2^2$ (the first row) and $\frac{g(t)^2}{2\sigma_t^2}\|\mathbb{E}_{q_t(x_t)}\left[\boldsymbol{\epsilon}_\theta^t(x_t)\right]\|_2^2$ (the second row) calculated on different datasets. The models on CIFAR-10 and CelebA is trained on discrete timesteps ($t = 0, 1, \cdots, 1000$), while those on AFHQv2, FFHQ, and ImageNet are trained on continuous timesteps ($t \in [0, 1]$). We convert data prediction $\boldsymbol{x}_\theta^t(x_t)$ into noise prediction $\boldsymbol{\epsilon}_\theta^t(x_t)$ based on $\boldsymbol{\epsilon}_\theta^t(x_t) = (x_t - \alpha_t \boldsymbol{x}_\theta^t(x_t))/\sigma_t$. The y-axis is clamped into $[0, 500]$.

Empirical Results

Table 1: Comparison on sample quality measured by FID ↓ with varying NFE on CIFAR-10. Experiments are conducted using a linear noise schedule on the discrete-time model from [15]. We consider three variants of DPM-Solver with different orders. The results with † mean the actual NFE is order × $\lfloor \frac{\text{NFE}}{\text{order}} \rfloor$ which is smaller than the given NFE, following the setting in [26].

Noise prediction	DPM-Solver	Number of evaluations (NFE) 10	15	20	25	30	35	40
$\epsilon_\theta^t(x_t)$	1-order	20.49	12.47	9.72	7.89	6.84	6.22	5.75
	2-order	7.35	†4.52	4.14	†3.92	3.74	†3.71	3.68
	3-order	†23.96	4.61	†3.89	†3.73	3.65	†3.65	†3.60
$\epsilon_\theta^t(x_t) - \mathbb{E}_{q_t(x_t)}[\epsilon_\theta^t(x_t)]$	1-order	19.31	11.77	8.86	7.35	6.28	5.76	5.36
	2-order	**6.76**	†4.36	4.03	†3.66	3.54	†3.44	3.48
	3-order	†53.50	**4.22**	†**3.32**	†**3.33**	**3.35**	†**3.32**	†**3.31**

Table 2: Comparison on sample quality measured by FID ↓ with varying NFE on CelebA 64×64. Experiments are conducted using a linear noise schedule on the discrete-time model from [35]. The settings of DPM-Solver are the same as on CIFAR-10.

Noise prediction	DPM-Solver	Number of evaluations (NFE) 10	15	20	25	30	35	40
$\epsilon_\theta^t(x_t)$	1-order	16.74	11.85	7.93	6.67	5.90	5.38	5.01
	2-order	**4.32**	†3.98	2.94	†2.88	2.88	†2.88	2.84
	3-order	†11.92	3.91	†2.84	†2.76	2.82	†2.81	†2.85
$\epsilon_\theta^t(x_t) - \mathbb{E}_{q_t(x_t)}[\epsilon_\theta^t(x_t)]$	1-order	16.13	11.29	7.09	6.06	5.28	4.87	4.39
	2-order	4.42	†3.94	2.61	†2.66	2.54	†2.52	**2.49**
	3-order	†35.47	**3.62**	†**2.33**	†**2.43**	**2.40**	†**2.43**	†**2.49**

Empirical Results

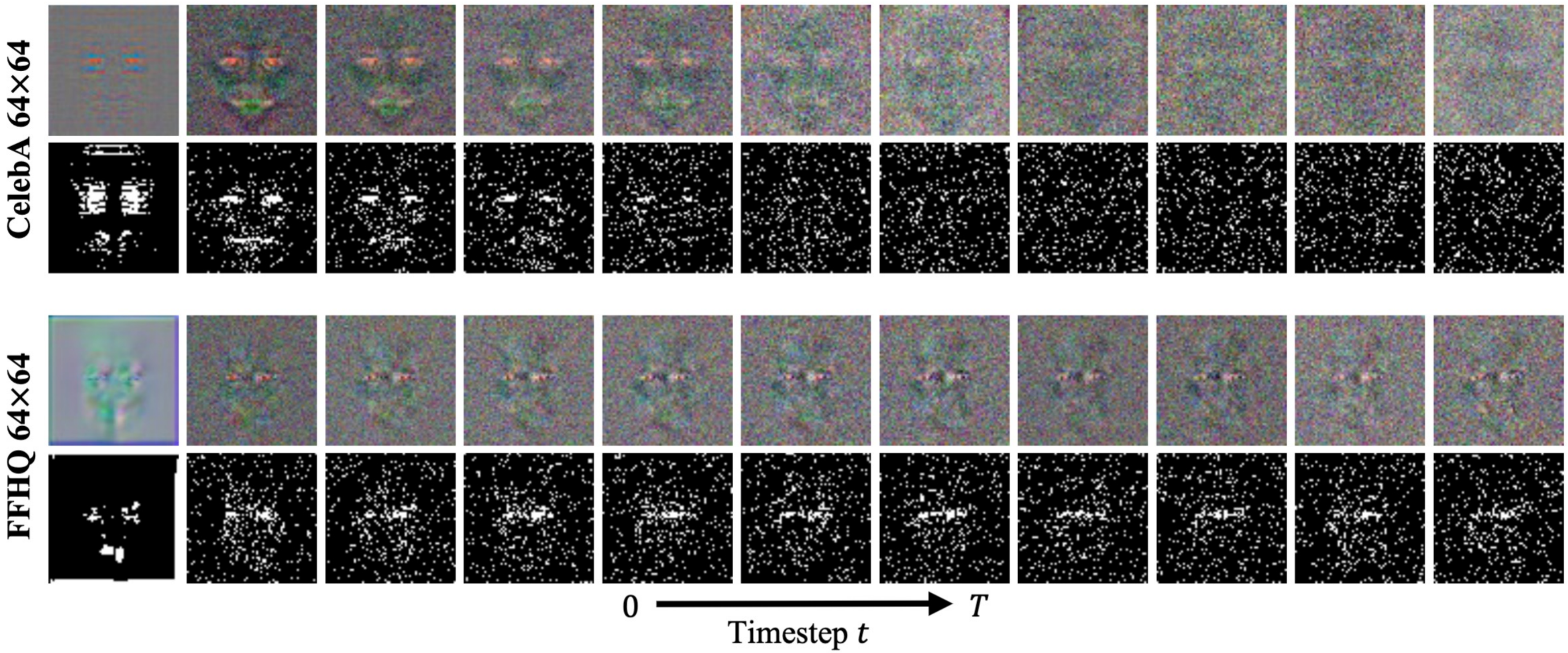

Figure 2: Visualization of the expected predicted noises with increasing t. For each dataset, the first row displays $\mathbb{E}_{q_t(x_t)}[\epsilon_\theta^t(x_t)]$ (after normalization) and the second row highlights the top-10% pixels that $\mathbb{E}_{q_t(x_t)}[\epsilon_\theta^t(x_t)]$ has high values. The DPM on CelebA is a discrete-time model with 1000 timesteps [35] and that on FFHQ is a continuous-time one [20].

Thanks